



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/607,895

06/27/2003

Barrett M. Kreiner

20009.0264US01(02242)

7209

45695 7590 06/06/2008
WITHERS & KEYS FOR BELL SOUTH
P. O. BOX 71355
MARIETTA, GA 30007-1355

EXAMINER

VU, TUAN A

ART UNIT

PAPER NUMBER

2193

MAIL DATE

DELIVERY MODE

06/06/2008

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/607,895	Applicant(s) KREINER ET AL.	
	Examiner Tuan A. Vu	Art Unit 2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 11 May 2008.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-10, 12 and 20-30 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-10, 12 and 20-30 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This action is responsive to the Applicant's response filed 5/11/2008.

As indicated in Applicant's response, claims 1, 20-27 have been amended, claims 11, 13-19 canceled, and claims 28-30 added. Claims 1-10, 12, 20-30 are pending in the office action.

Claim Objections

2. Claim 25 is objected to because of the following informalities: the claim has been marked as "Original" yet contains added text modifications. This improper marking, according to CFR § 1.121(c) would have been categorized a non-compliant response, but instead has been treated herein as a mere objected to impropriety. Future observation for compliancy to this claim-marking rule is required.
3. The inconsistent use of 'first computing system' to mean 'first computer system' requires that correction throughout the claims (e.g. claim 20).

Specifications

4. The disclosure is objected to because of the following informalities: some teachings described in (i) para 0031, pg. 11, (ii) para 0034, pg. 12, (iii) para 0035, pg. 13, (iv) para 0038, pg. 14 amount to misuse of syntax leading to inaccurate semantic that would not be consistent with the corresponding context being described (e.g. Fig. 2-3), as following:

(i) the translator 222a on the OS 218a of the client is taught as processing XML transmitted from server 112a to yield native instructions that can be processed on the client machine. The phrase "that can be processed by OS 210" (at end of the paragraph) is deemed a unacceptable semantic in that native format translated at a client OS *can be processed by a*

remote OS like OS 210, when normally, native code cannot be compatible to both server and client platforms, hence the need for XML conversion as shown in Figure 3.

(ii) The phrase 'In server system 116c' (top line of page 12) appears to describe client system 116c with remote access module 220c, not a server system; hence correction is required.

(iii) The phrase 'At step 316, OS 210 may execute ... so that output is generated ... system 112 as well as at client ... system 116' (bottom of 1st paragraph) is not commensurate with step 316. The execution by a server OS to yield output data at system 116 (i.e. without any XML transmission) during step 316 is not falling within the purport of step 316, whereby request for output is translated at the server into a XML format which then can be transmitted to client 116, no execution of native code disclosed therein.

(iv) The phrase 'the instructions are executed by OS 210a' appears to be a misrepresentation of step 324, because XML conversion into a native format deemed compatibly executable only at client 116a, cannot also be described as instructions being executed by server OS 210a within the context of step 324, which is subsequent to receiving XML data from server 112a.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

6. Claim 26, 29 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not

Art Unit: 2193

described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Specifically, claim 26 recites 'at the second computer, translating ... from the non-proprietary format ... second user input instruction ... ; executing ... by the second computer' then 'receiving data from the second operating system related to the second user input... data defining first system output instruction ... relating to first input instruction ... incompatible with the first operation system'; then 'creating ... a second software object in ... non-proprietary format' then 'transmitting ... second software object from the second computer to the first computer'. The above step (underlined) of *receiving from* the second operating system of data defining first output instruction compatible with the second operating system but not with the first operating system' is deemed nowhere described in the scenario represented by Figure 2-3 of the Specifications, because the output instruction data **being non compatible with the first operating system** is not disclosed as being **received from the second operating system** anywhere **by any machine**; nor is this received 'first output instruction' (from the second operating system) disclosed as being **prior to** creating and **transmitting** of XML object back (from the second operating system, as claimed) to the first operating system, i.e. as received, this 'first output instruction' comes **from** the second operating system. The Specifications (refer to Figure 3) as a whole does not teach a intermediate step where output instruction (related to user input) being executable only at the server end is received **from the server itself**; that is, after a server executing of proprietary code at the server, a 'first output data instruction' not compatible with the first client machine **is received from the server prior to** the server (after translating

Art Unit: 2193

output data into and) creating of a XML object for transmission and directed for being processed at the client operating system.

The above 'receiving data from the second operating system' will be treated as generating output instruction related to first input user input instruction at the server end prior to the server translating output data into a XML object to be retransmitted back to the first operating system.

Claim 29 fails to cure to the above lack of description, hence is also rejected for the reason that the inventor has no possession of the (above underlined) limitation at the time the invention was made.

Claim Rejections - 35 USC § 102

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

8. Claims 1-10, 12, 20-30 are rejected under 35 U.S.C. 102(e) as being anticipated by Salmenkaita et al., USPN: 2004/0176958 (hereinafter Salmenkaita).

As per claim 1, Salmenkaita discloses a method for providing remote computer control of an application executing on a second computer from a first computer over a network, comprising:

via a first user interface of the first computer, receiving a first user input instruction by a proprietary operating system on the first computer for execution, the first user input instruction

Art Unit: 2193

being operationally compatible with the proprietary operating system (*voice command* – Fig. 2A, 2D; receive voice command 282 – Fig. 4I; Fig 5A; user input 710-Fig 7A, input 730 – Fig. 7B; Fig. 4C-4D – Note: input commands at client reads on being compatibly proprietary to the user machine) and operationally incompatible with a second operating system executing on the second computer (Note: voice input at client interface reads on operationally not used at server machine for direct execution thereat) which incorporates a second user interface, wherein the first user interface is dissimilar to the second user interface;

at the first computer, translating the first user input instruction into a non-proprietary data script defining at least one XML item utilizing a first device driver (Fig. 3A; para 0172-0174, pg. 14; para 0167, pg. 13) resident in the proprietary operating system on the first computer, wherein the first device driver formats the first user input instruction into at least one XML item (Fig. 3A; para 0167, pg. 13; *voice XML tags* -para 0052; *embed voice tags in a XML message* -- para 0056-0061, pg. 4-5; para 0172-0174, pg. 14; para 0232 pg. 19) corresponding to the first user input instruction;

transmitting the non-proprietary data script defining the at least one XML item from the first computer to the second computer (para 0085-0086, pg. 8; *Message 515, XML file 227* - Fig. 4C, D);

translating the non-proprietary data script defining the at least one XML item into a second user input instruction utilizing a second device driver in the second operating system on the second computer (e.g. step 736 – Fig. 7B; para 0258, pg. 21; steps 216, 240, 242, 244, 246 – Fig. 4D), wherein the second device driver translates the at least one XML item corresponding to the first user input instruction into the second user input instruction, the second user input

Art Unit: 2193

instruction being compatible with the second operating system on the second computer and incompatible with the proprietary operating system on the first computer (xml 227 - Fig. 4c, 4d; services 440, 442, 444, 446, 448, 450 method calls – Fig. 6; *invoke ...method ... metadata vector* – para 0258, pg. 21--Note: server with proprietary services to effect recommendations fulfilling applications whose results are sent back to client wireless reads on not compatible with native environment of wireless client – see Fig. 5A), the second user input instruction being functionally similar to the first user input instruction for execution on the second computer (e.g. boxes 216, 240, 242, 244, 246 – Fig. 4D; para 0249, pg. 21; steps 364-366 Fig. 5A); and executing the second user input instruction on the second computer (e.g. boxes 216, 240, 242, 244, 246 – Fig. 4D; para 0177, pg. 15; Fig. 4E; para 0225-0227, pg. 18; Fig. 5; *receive ... service 368* – Fig. 5A; Fig 7A-B; method calls – Fig. 6).

As per claims 2-3, Salmenkaita discloses wherein receiving said first user input instruction comprises receiving an instruction for outputting data or displaying data (e.g. display area 102B --Fig. 1; *recommended services* – Fig 2B-C; Figs. 3; *prepared updated MENU 224* – to device 100: *MENU message 509* – Fig. 4B, 4D – Note: selection by wireless user for a recommendation being serviced and updated by server for retransmission back to wireless client as updated recommendation MENU reads on instruction of data outputting)..

As per claim 4, Salmenkaita discloses receiving an instruction for outputting data which further comprises receiving an instruction for generating a sound (e.g. *audio metadata 125'* – Fig. 4B; *audio output* - para 0085, pg. 8).

As per claims 5 and 7, Salmenkaita discloses receiving said first user input instruction which comprises receiving an instruction for inputting data; an instruction indicating a computer keyboard input (Fig. 1).

As per claim 6, Salmenkaita discloses HW input receiving via a touch pad, the use of touchpad in some small device to provide mouse functionality was equivalent to a mouse click (touch pad as in *Touch sensor* - para 0072, pg. 6; Fig. 1).

As per claim 8, Salmenkaita discloses wherein translating the first input instruction into a data script defining at least one XML item comprises generating a first XML tag defining the beginning of the XML item, generating a data item corresponding to the first input instruction, and generating a second XML tag defining the end of the XML item (e.g. Table D, E, pg. 14; para 0155, pg. 11; *processing instruction* – para 0163-0164, pg. 12).

As per claim 9, Salmenkaita discloses transmitting the data using HTTP (e.g. Fig. 6, para 0179, pg. 15; para 0266-0271, pg. 22; Fig. 3D).

As per claim 10, Salmenkaita discloses wherein translating the data into a second instruction comprises identifying a first XML tag defining the beginning of an XML item, identifying a data item corresponding to a input instruction, identifying a second XML tag defining the end of an XML item (para 0232, pg. 19; *specification ... activity* – para 0156, pg. 11; para 0163-0164, pg. 12).

As per claim 12, Salmenkaita discloses a computer readable medium (refer to claim 1 for corresponding rejection) having computer-implementable instructions stored thereon for performing the method recited in claim 1.

As per claim 20, Salmenkaita discloses a system for remote computer access between computing systems with incompatible operating systems, comprising:

a first computing system having stored thereon software which when executed on the first computing system;

receives a user input via a first user interface of the first computing system (e.g. Fig. 4C; para 0051, pg. 4);

identifies user input instructions generated by a proprietary operating system on the first computer system (*voice command* – Fig. 2A, 2D; receive voice command 282 – Fig. 4I; Fig 5A; user input 710-Fig 7A, input 730 – Fig. 7B; Fig. 4C-4D), input instructions relating to generating a system output (send information 366, Fig. 5; Fig. 6 and *Visit Object* of para 0252,0255, 0258, pg 21) via a second user interface (para 0272, pg. 22; browser - para 0279-282, pg. 23; browser 102 - Fig. 3A-C – Note: returned results from WAP servers to be displayed in mobile browser reads on output via a second interface of first computing system) of the first computing system in response to the user input;

translates the user input instructions into a first non-proprietary data script defining an outgoing software object corresponding to the user input instructions (e.g. para 0052-0061, pg. 4-5; voice tags - para 0172-0174, pg. 14); the translation being accomplished by a first device driver executing in conjunction with the proprietary operating system on the first computer system,

transmits the outgoing software object (message 515 -Fig. 4C, D), and

receives an incoming software object comprising a second non-proprietary data script (Fig. 4D; *message 515 ... to the network server 140* - para 0166-0167 pg. 13; para 0232, pg. 19)

Art Unit: 2193

reflecting a response to the user input instructions (step 228 Fig. 4C) for execution on the second user interface of the first computing system (e.g. para 0272, pg. 22; browser - para 0279-282, pg. 23; recommendations - Fig. 3A-C); wherein the second non-proprietary data script is translated by the first device driver into a system output instruction being compatible with the proprietary operating system of the first computing system (para 0272, pg. 22; browser - para 0279-282, pg. 23; recommendations - Fig. 3A-C) and incompatible with a second computing system (Note: results from servers being displayed as browser recommendations for user to evaluate reads on browser-specific rendered output by the WAP mobile device only), the system output instruction then being executed on the first computer system as a system output via the second user interface.

As per claim 21, Salmenkaita discloses a method for providing remote computer access, comprising:

receiving outgoing instructions (e.g. *voice command* – Fig. 2A, 2D; receive voice command 282 – Fig. 4I; Fig 5A; user input 710-Fig 7A, input 730 – Fig. 7B; Fig. 4C-4D) relating to generating an output on a first computer from a first operating system executing on the first computer, the instructions being compatible with the first operating system and incompatible with a second operating system on a second computer;

creating data defining a first XML item corresponding to the outgoing instructions relating to generating output on the first computer, wherein the instructions are translated into the first XML element at the first computer (Fig. 3A; para 0167, pg. 13; *voice XML tags* -para 0052; *embed voice tags in a XML message* -- para 0056-0061, pg. 4-5; para 0172-0174, pg. 14; para 0232 pg. 19);

transmitting the first XML element from the first computer to the second computer (e.g. para 0085-0086, pg. 8; *Message 515*, *XML file 227* - Fig. 4C, D);

receiving data defining a second XML item in response to the outgoing instructions to generate output on the first computer from the second computer (e.g. XML 250, Fig. 4G; XML 250 – Fig. 4E);

creating incoming instructions relating to generating the output from the data defining the second XML item (XML 250 – Fig 4G), wherein the incoming instructions are translated from the second XML item at the first computer (display 256 – Fig. 4G; recommendations - Fig. 3A-C; step 268, 270, 272 – Fig 4H; steps 254, 256, 258 – Fig. 4G) and after which are compatible with the first operating system on the first computer and incompatible with the second operating system on the second computer; and

executing the incoming instructions to generate the output at the first computer relating to inputs on the second computer (e.g. para 0272, pg. 22; browser - para 0279-282, pg. 23; recommendations - Fig. 3A-C; step 268, 270, 272 – Fig 4H; steps 254, 256, 258 – Fig. 4G).

As per claim 22, Salmenkaita discloses wherein receiving incoming instructions relating to generating output comprises receiving instructions relating to generating visual or audio output (e.g. para 0272, pg. 22; browser - para 0279-282, pg. 23; recommendations - Fig. 3A-C; para 0280, pg. 23).

As per claim 23, Salmenkaita discloses wherein creating the first XML item corresponding to the outgoing instructions relating to generating output comprises generating at least a first XML tag defining the beginning of the first XML item, generating a data item corresponding to the instruction relating to generating output; and generating at least a second

XML tag defining the ending of the first XML item (e.g. para 0173-0174, pg. 14; Table D, E, pg. 14; para 0155, pg. 11; *processing instruction* – para 0163-0164, pg. 12).

As per claim 24, Salmenkaita discloses wherein transmitting the data defining the first XML item comprises transmitting the data defining one the first XML item using HTTP protocol (e.g. Fig. 6, para 0179, pg. 15; para 0266-0271, pg. 22; Fig. 3D).

As per claim 25, Salmenkaita discloses wherein creating incoming instructions relating to generating the output ((para 0272, pg. 22; browser - para 0279-282, pg. 23; recommendations - Fig. 3A-C) comprises identifying a first XML tag identifying the beginning of the XML item, identifying a data item corresponding to an input, and identifying a second XML tag identifying the ending of the XML item (e.g. XML 250, Fig. 4G; XML 250 – Fig. 4E).

As per claim 26, Salmenkaita discloses method for providing remote computer access between computing systems with incompatible operating systems, comprising:

receiving a first user input instruction relating to a user input received via a first user interface of the first computer by a first operating system on the first computer, the first user input instruction being compatible with the first operating system and incompatible with a proprietary second operating system on the second computer (refer to claim 1);

creating data defining a first software object in a non-proprietary format (Fig. 3A; para 0172-0174, pg. 14; para 0167, pg. 13) corresponding to the first user input instruction relating to the user input;

transmitting the first software object from the first computer to the second computer (e.g. para 0085-0086, pg. 8; *Message 515, XML file 227* - Fig. 4C, D);

at the second computer, translating the first software object from the non-proprietary format to a second user input instruction compatible with the proprietary second operating system and incompatible with the first proprietary operating system (boxes 216, 240, 242, 244, 246 – Fig. 4D; boxes 240, 243, 244, 246, Fig. 4F); executing the second user input instruction by the second computer (e.g. boxes 216, 240, 242, 244, 246 – Fig. 4D; para 0177, pg. 15; Fig. 4E; para 0225-0227, pg. 18; Fig. 5; *receive ... service 368* – Fig. 5A; Fig 7A-B);

receiving data from the second operating system related to the second user input instruction being executed, the data defining first system output instruction, the first system output instruction relating to the first user input instruction and being compatible with the second operating system executing on the second computer and incompatible with the first operating system on the first computer (Note: this step is treated as output data instruction related to first user input instruction is generated at server prior to transmission of data back to the client - boxes 216, 240, 242, 244, 246 – Fig. 4D; para 0177, pg. 15; Fig. 4E; para 0225-0227, pg. 18; Fig. 5; *receive ... service 368* – Fig. 5A; Fig 7A-B);

creating data defining a second software object in the non-proprietary format that corresponds to the second user input instruction (XML 250, Fig 4E, 4G, 4F);

transmitting the second software object from the second computer to the first computer (XML 250, Fig 4E, 4G, 4F);

at the first computer, translating the second software object to a second system output instruction being compatible with the proprietary first operating system and incompatible with the proprietary second operating system; and executing the second system output instruction to render the user output by the first computer on a second user interface (para 0272, pg. 22;

browser - para 0279-282, pg. 23; recommendations - Fig. 3A-C; step 268, 270, 272 – Fig 4H; steps 254, 256, 258 – Fig. 4G).

As per claim 27, Salmenkaita discloses wherein transmitting the data defining the first and second software objects at least one XML item comprises using the HTTP protocol to transmit the first and second software objects data defining at least one XML item (e.g. first object: XML 235, 231, Fig. 4F; second object: XML 250, Fig 4G).

As per claim 28, Salmenkaita discloses wherein the first user interface (e.g. input 264, Fig. 4H; microphone 103, keypad 104 touch sensor audio sensor light sensor - Fig. 3) is different from the second user interface (step 256 – Fig. 4E; browser 102, Fig. 3B).

As per claim 29, refer to claim 28

As per claim 30, Salmenkaita discloses the system for remote computer access between computing systems with incompatible operating systems of claim 20, further comprising

the second computing system having stored thereon software which when executed on the second computing system:

receives the outgoing software object from the first computing device (XML 227 Fig 4D; XML 235 Fig. 4F); translates the first non-proprietary data script using a second device driver executing in conjunction with a second proprietary operating system executing on the second computer system into the user input instructions identified by the first computing system but operationally compatible with a second operating system executing on the second computer system (steps 242, 244, 246 – Fig. 4D; step 243, 244, 246 – Fig. 4F; Method calls – Fig. 6) and operationally incompatible with the proprietary operating system executing on the first computer system;

executes the user input instructions compatible with the second operating system (steps 242, 244, 246 – Fig. 4D; step 243, 244, 246 – Fig. 4F; Method calls – Fig. 6; Fig 7A-B); identifies system output instructions operationally compatible with a second operating system executing on the second computer system and operationally incompatible with the proprietary operating system executing on the first computer system, the system output instructions being responsive to the user input instructions identified by the first computing system (Note: method calls performed at client machine reads on identifying of second operating system instruction related to required output in response to requirements of incoming XML received from mobile device 100 - see Fig. 4D, 4F; Fig. 6),

translates the system output instructions into a second non-proprietary data script defining an incoming software object utilizing the second device driver transmits the incoming software object (XML 250 – Fig 4G, 4F); and

a communications network operably coupled between the first computing system and the second computing system for transmitting the first and second non-proprietary data scripts defining incoming and outgoing software objects between the first computing system and the second computing system (XML 250 – Fig 4G, 4F).

Response to Arguments

9. Applicant's arguments filed 5/11/08 have been fully considered but they are not persuasive. Following are the Examiner's observation in regard thereto.

USC § 102 Rejection:

(A) Applicants have submitted, regarding Salmenkaita, that just because operating systems of the wireless device and the server require translation of voice into machine language format that

is compatible with their operating system, it does not follow that Salmankaita describes that wireless device OS and network server OS are incompatible with each other (Appl. Rmrks pg. 15, middle).

First, the claim language depicts incompatibility in terms of 'input instruction' being incompatible with operating system of second system (re claim 1). There are no explicit specifics between respective operating system of first computer system and second computer system in the claim in order to derive what is considered compatible or incompatible respective to each of OS of first and second computer system as claimed. The claimed 'incompatible with a second operating system' is analyzed and necessarily interpreted in conjunction with the context relating a 'input instruction' with 'second operating system'; and in broad interpretation, a voice input or keyboard input intended for a architecture like a mobile device is deemed incompatible with the OS of a WAP server, which Salmenkaita teaches. That is, Salmenkaita does not teach a network server 140 (Fig. 5) receiving a XML message (Fig. 4F) as being actually a wireless device equipped with architectural OS with the same keyboard/microphone input interface or voice sensors identical to those of wireless device 100 (Fig. 3A).

Second, in terms of definition of 'operating system' and 'incompatible', the claim does not redefine *operating system* or *incompatible* in a more distinguishing way other than normal understanding ordinarily attributed to these terms. The Specifications mention about format of input or instructions that require re-translation; but nowhere is there a section where the terminology of 'incompatible' and 'operating system' is defined explicitly and deliberately in a particular manner to preclude standard interpretation from being applied. Thus, operating system in a broad sense is analogized to the very architectural OS of a machine including software and

hardware, like that of a PC or a PDA equipped with its ports, registers, bus system, CPU, kernel and underlying native engine or system calls designed for a given platform architecture. For example, the OS of a WAP mobile device and the architecture of a remote server machine, each of which equipped with different instruction architectural, different bus and kernel, registers, or system routines, application interfaces or DLL. Format of a voice data or a keyboard input entered at a wireless device interface cannot be deemed compatible with the OS of a server, which is not designed with port to dynamically acknowledge that format and provide native code execution. The teachings in Salmenkaita whereby KB or microphone inputs are to be translated in order to be parsed at the server 140 (see Fig 4, Fig. 5A) teach how format is incompatible in a sense that it requires conversion into a XML message.

Third, in a whole, the claim entails an input instruction that is to be translated into a non-proprietary form because of the fact that its original initial format it cannot be understood by the second operating system. The claim does not provide a scenario in which two operating system exhibit dire differences in order for the claim to be addressed only to the extent of comparing what particularly distinguish first and second operating system. Even in light of the Specifications, 'incompatible' only means that some initial format is to be retranslated into a less proprietary form for another machine equipped with means to interpret this more neutral format to interpret the meaning of the original format. Salmenkaita teaches (i) use of a neutral XML form to convey the wireless machine original user input to a WAP server for this server to derive the content of this message and address the directives of the wireless device original request (e.g. for more recommendations); and as such, Salmenkaita has fulfilled the limitation as to input instruction being 'incompatible' with second operating system (e.g. claim 1). The use of neutral

format (ii) to resolve discrepancies issues because of operating system is also observed in the current Applications (see pg. 16, para 0044, top para). Hence, in light of this recourse to XML reconversion as construed from para 0044, Applicants' alleging that server platform in Salmenkaita is NOT INCOMPATIBLE with the wireless device 100 (as cited in the rejection) cannot be in agreement with the above similarity of teachings as in (i) and (ii). However, the claim only requires that input instruction not being compatible with second operating system, which has been addressed above.

Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the reference.

The argument is non persuasive.

(B) Applicants have submitted that the allegations by the Office Action that server and wireless machine in Salmenkaita are 'mutually incompatible' seem to flow from implicit teaching (as in inherency), which is not commensurate with a MPEP 2131 where extrinsic evidence is required (Appl. Rmrks pg. 16, top half). The claim language has been analyzed based on 'input instruction' from a first machine is being incompatible with respect to proprietary system of another machine, and a non-proprietary form is created to convey this first machine generated input instruction to the second system. The claimed 'incompatible' in particular has been addressed with Salmenkaita as set forth above; and as a whole, the paradigm (e.g. claim 1) for using non-proprietary form to overcome incompatible issue because of input instruction format has been constructed with broad reasonable interpretation and fulfilled by the similar paradigm from the reference, according to the explanation as set forth in section A. There is no

statement effectuated in the Office Action in addressing these limitations to the effect that ‘inherent’ teaching has been applied or should be integrated. The argument is not commensurate with the grounds of rejection.

(C) Applicants have submitted that untranslated voice command is translated by the server for matching it to voice cuts (Appl. Rmrks pg. 16, bottom) making it clear that Salmenkaita’s operating system in server is compatible with operating system of wireless device. The voice commands are not disclosed to be translated raw in the communications between wireless device 100 and server device 140 all throughout Salmenkaita’s disclosure, because voice commands are translated into metadata or tagged information (i.e. XML format) interpretable only by W3C type of interpreter, for which the rejection has provided with extensive cited Figures, and text. The ‘incompatible’ argument has to be referred back to section A.

(D) Applicants have submitted that based on remarks made from a previous telephone interview there is an Examiner’s ‘predicate assumption’ that it must be a incompatibility issue yet to be addressed (Appl. Rmrks pg 17, bottom). This type of remark is not a provision of facts in order to point out how the a particular claim language distinguish over a cited portions as effectuated by the Office Action. What the Applicants perceived as a ‘predicate assumption’ amounts to no legal grounds as to substantially refute the validity of the evidences proffered in the Office Action in regard to the 102 rejection; that is, not only this recognition on a ‘predicate assumption’ fails to constitute a legal admission made by the Office when rejecting that particular language but also is deemed devoid of supporting evidences which otherwise are required for a proper prima facie in respect to a CFR § 1.111b rule.

The claims as submitted stand rejected as set forth in the Office Action.

Conclusion

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (571) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lewis Bullock can be reached on (571)272-3759.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence - please consult Examiner before using) or 571-273-8300 (for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Tuan A Vu/

Primary Examiner, Art Unit 2193

June 3, 2008

